Online Help

<u>Overview</u>

Online Help is a very important part of the user interface. As a programmer, you probably will not be responsible for writing the Help. However, you're likely to be responsible for providing information that technical writers will use to author the Help system. Delivering a quality Help system requires significant input and review from the application developer. You, the developer, are the one who really knows how the program works, and if you have followed a solid user interface design process, you should have a good understanding of the user tasks that are supported by the program.

Think of Help as user assistance. When users are confused on how to complete their task while using an interface, they need information that will answer their questions and clear their confusion. This information differs from other forms of online documentation in that it is delivered at the user's request. The content can be procedural, contextual, tutorial, or reference. Online Help systems have gone through significant style changes lately. No matter what the content, Help systems should be simple, efficient, and relevant so the user can get the help they need without getting lost in the user interface. The goal of a Help system should not be to teach users about the system's capabilities and functions, but rather to provide quick and immediate access to information about a specific task, command, or message.

Today's Help systems are moving away from the old-style Help which was almost entirely reference based. Old-style Help typically was displayed as a near full screen window, and it included so much information that the user had to force themselves to read it to find the one piece of information they were looking for. This is in sharp contrast to modern *task-oriented Help* where the focus is on the user tasks. The figure titled, *Task-oriented Help*, illustrates the focus that is now placed on the user's task. The task-oriented style of Help has the following characteristics.

- Topics are short and task-oriented. Large Help subjects are divided into smaller tasks.

- Help windows are small so that the user can look at the program while reading the Help text.

- Hyperlinks are used to connect to relevant information.

Task-oriented Help
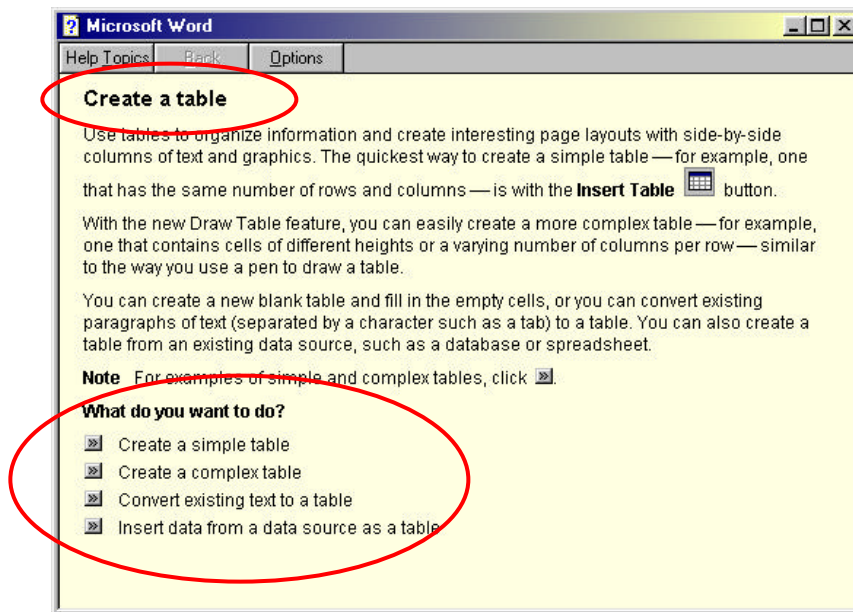
Context-sensitive help supports the user gaining information about a particular object or element on the screen, including controls that are in property sheets and dialog boxes. It answers the questions of "What is this?" or "Why would I want to use it?" The figure titled, *What's This? Help*, indicates the ways it can be accessed.

What's This? Help
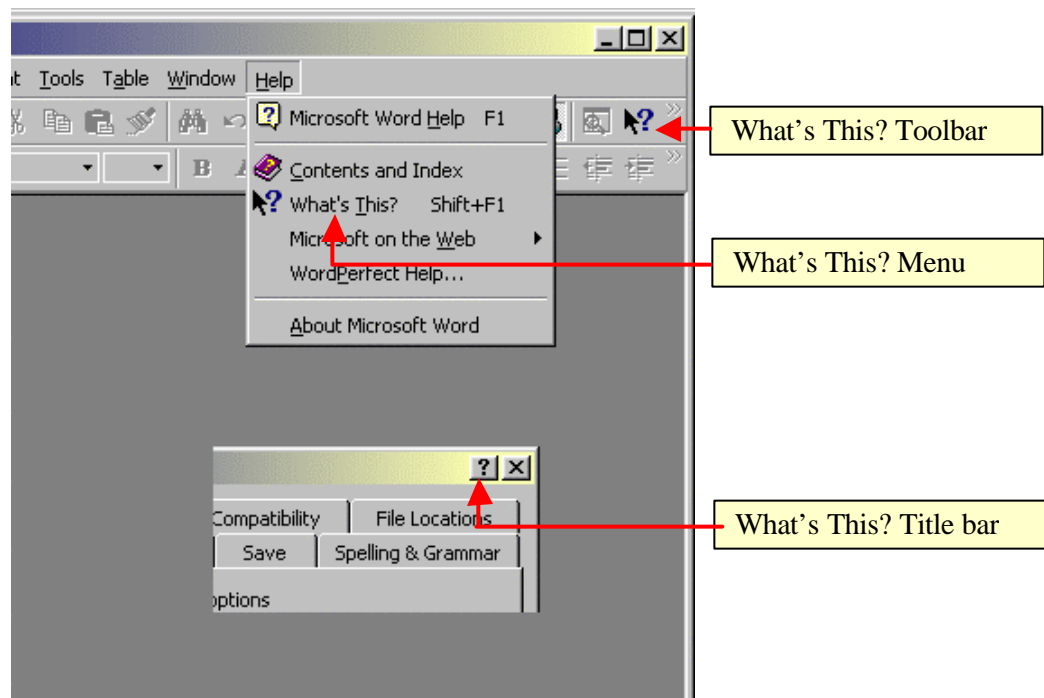
Contextual Help can also be provided for a property sheet, dialog box, or a message box by including a Help button in the window as shown in the figure titled, *Secondary Window Help Button.*

Secondary Window Help Button

Developers should not write a help system on their own. Use of a technical writing staff is highly recommended. Technical writers have great skill in bringing words to the online medium. Developers have to work with technical writers to get them to understand how to use the application better than the typical user reading the documentation. As the Help project progresses, the developer must keep the technical writer informed of significant changes to the application, and review intermediate results.

The goal of this section of the style guide is to offer guidance on incorporating Help into a Windows style program, and recommending specific items to check to make sure that Help is done correctly. This document's primary focus is on Windows style applications. However, the web is having an impact on the Help interface of Windows style applications. This can be seen in the newer HTML Help Browser that is illustrated in the figure titled *HTML Style Help Browser*. This Help browser, which was included in Windows 98 and Windows 2000, differs from the WinHelp browser that was included in prior versions. While Windows 98 and Windows 2000 still include this support for backward compatibility, HTML Help is a better choice. HTML Help uses common Web conventions, providing a familiar and consistent way for users to learn to navigate through Help information. You can use HTML Help to support Help interfaces for both conventional and Web-style applications. It also enables you to easily include local Help topics as well as HTML pages on Web sites.

HTML Style Help Browser
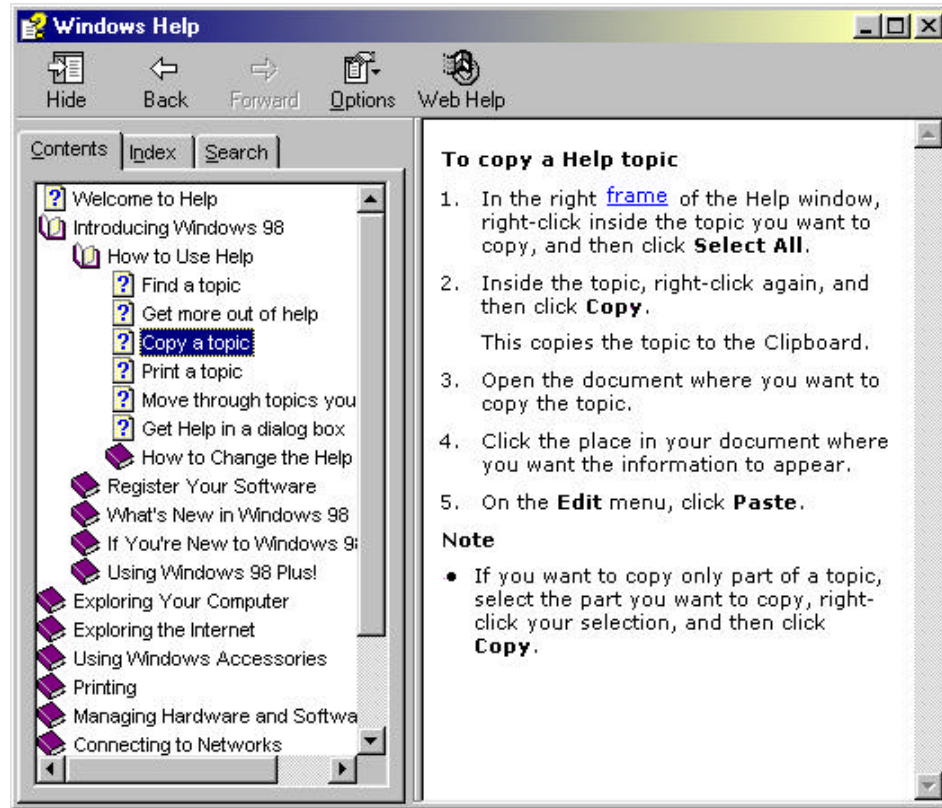
Remember that users just want to get their work done. Stay focused on two very important goals when putting together the Help system:

1. Provide users with the steps they need to get their work done as quickly as possible

2. Make the information easy for users to find.

General Help Standards

The following Help standards are to be adopted across all CDC Windows and web based surveillance applications. These standards should lead to a more consistent and usable interface.

Help menu will follow CDC standards and include:

- Content and Index
- What's This?
- About

CDC surveillance applications will have Help menus. The Help menu will contain, at a minimum, menu items to initiate the Help browser on content and

4

index, the What's This? help menu item to initiate context-sensitive, and the About screen as shown in the figure titled *Help Menu Standards*.

Help Menu Standards
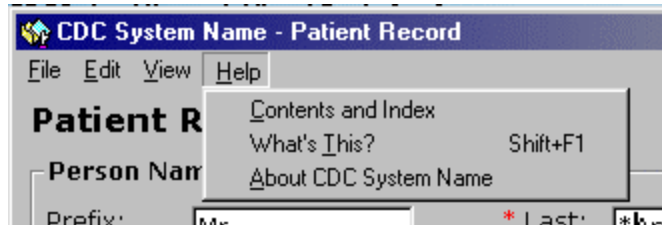
Keep Help windows small.

Users need to be able to see the interface that they are working in while reading the Help. Therefore, minimize the size of the Help window so that the user can continue to work in the program interface while repositioning the Help window.

Do not override F1 or Shift+F1.

Help is always accessed with F1, while context-sensitive Help is always accessed using Shift+F1. These hot keys are a Windows standard and should not be overridden or used for anything other than Help. For more information, see the Navigation and Metaphor section of this style guide.

Provide contrast between application window and the procedureal help window.

- *Standard yellow for WinHelp browser*
- *White for HTML Help browser*

In the windows environment where procedural Help windows will lay on top of application windows, it is important to provide contrasting color between the Help procedure and the application. Help windows should contain the standard yellow background as seen in all Windows Help procedural windows that use the WinHelp browser.

*Support context sensitive help whenever possible*.

Context-sensitive or What's This? Help is the most commonly encountered form of user assistance. Context-sensitive Help provides brief descriptions about the purpose of a control or other object in the interface. Because this form of on-demand user assistance is provided throughout the interface, users come to depend on its availability.

Provide What's This? Help on all property sheets and dialog boxes.

Property sheets and dialog boxes are not described with Help topic windows, rather they are described with mouse-driven context-sensitive Help using the What's This? Command as shown in the figure titled *Property Sheet Help*.

Property Sheet Help
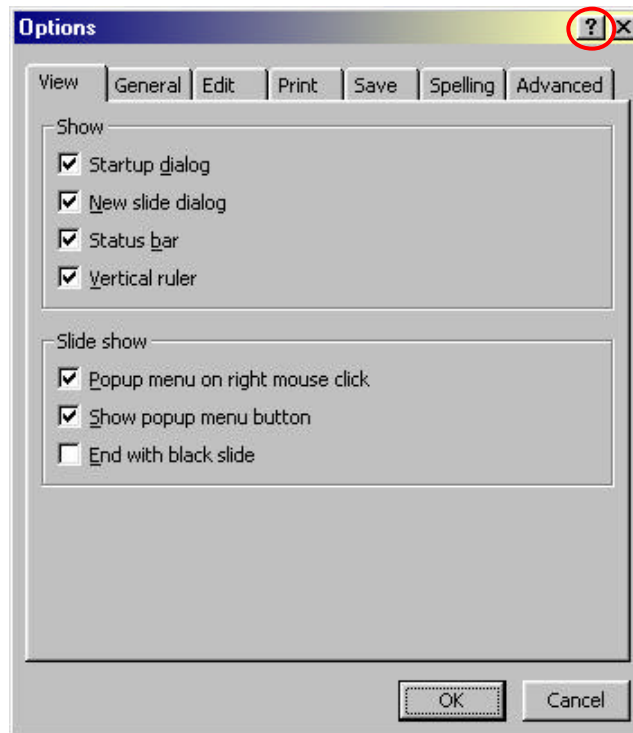
Make graphics of interface elements flat.

If you show a graphic of a button or other interface element in Help, make the graphic flat as shown in the figure titled *Help Graphic Example*. In this way, it doesn't look like a button that a user can click on. However, shortcut buttons and related topic expanders can be beveled like a button because they result the Help system performing some action.
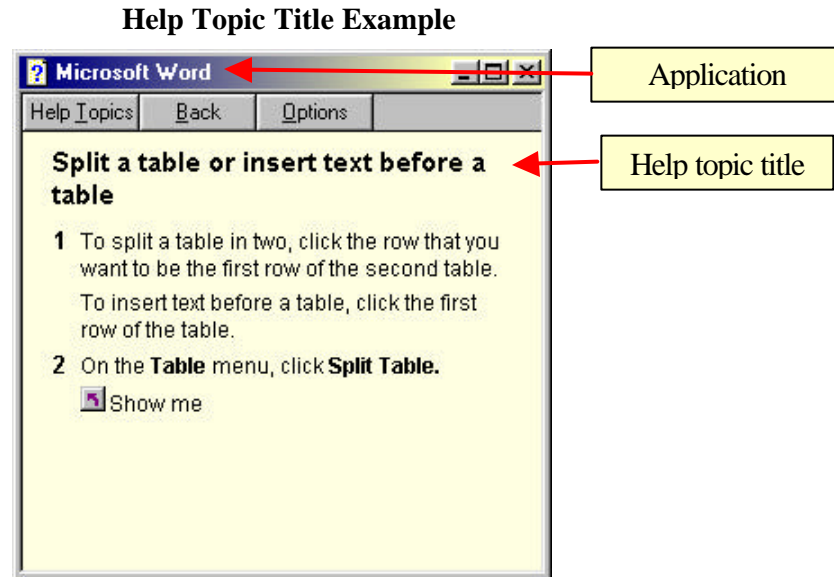
Help Graphic Example
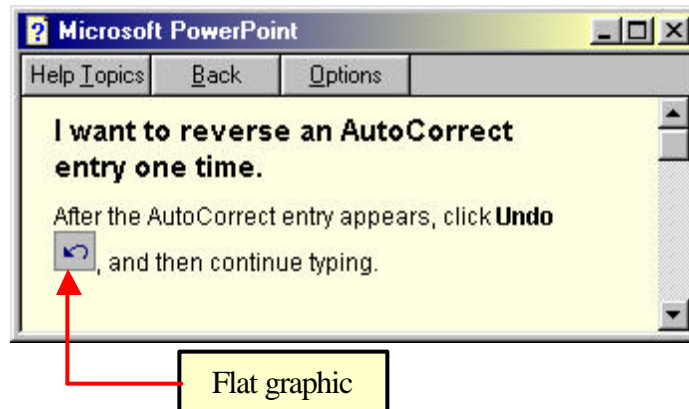
Do not include "click OK" in the last step of a procedure.

Usability studies show that users can complete the task without this step. Include "click OK" only if the user has to click OK in the middle of a procedure before continuing to the next step.

*Include **topic titles in all Help topics***.

All Help topics should include a topic title that is no more than two lines shown as illustrated in the figure titled *Help Topic Title Example*. The title identifies the topic and provides the user with a landmark within the Help system. The title should correspond to the entries included in the Help browser window.

Help Topic Title Example

![Microsoft Word Help window — Application label points to title bar, Help topic title label points to "Split a table or insert text before a table"]

Include CDC system name in title bar of all Help windows.

This avoids confusion between the Help for the current application and Help for any other application that may be active.

Provide consistent terminology.

As with writing any document, using consistent terminology is extremely important. Make sure you go through a process of identifying by name all the elements of the interface, user tasks, etc. Call them by the same name consistently throughout the Help procedures, conceptual topics, reference topics, and context-sensitive topics. In addition, avoid the use of pronouns. They tend to confuse the reader.

Write clear and concise index entries.

Microsoft makes the following recommendations for writing index entries:

- Keywords should be lowercase unless they are proper nouns (for example, use "files" and "File menu").
- Nouns should be plural unless the singular form is more correct (for example, use "programs" but not "Start menus").
- Verbs should be in the gerund form (for example, use "copying files," not "copy files").
- Keyword indexes should have no more than two or three levels of indentation.

General Help Guidelines

The following guidelines have been established within the user interface design community and should be applied across CDC software and web-based surveillance applications using HTML.

Apply Help index keywords to topics using the following considerations:

- Novice users
- Advanced users
- Common synonyms
- Words that describe the topic generally
- Words that describe the topic discretely

Keep Help topics short.

Procedural topics contain information about how to complete a task. Usability studies show that users are more likely to read a Help topic if it is short. In order to keep the topics as short as possible follow these guidelines:

- Consider what the user might need to accomplish, not how the software works.
- Consider the questions a user would ask when something goes wrong and write procedures to solve a single problem.
- Don't provide a lot of choices. Users don't want choices when they open a Help topic. They just want to get their task done. Once users are familiar with the application, they may be interested in learning more efficient ways to work, but these techniques can be included in "Tips and Tricks".
- Include one procedure per topic. If procedures are related, link them in a related topics jump.
- Point users to context-sensitive Help for detailed help on dialog box options. There's no need to duplicate information.
- Use shortcut buttons in procedures that users will carry out only once or infrequently as shown in the figure titled *Help Procedure*.
- Keep topics short enough so users don't have to scroll.

Help Procedure
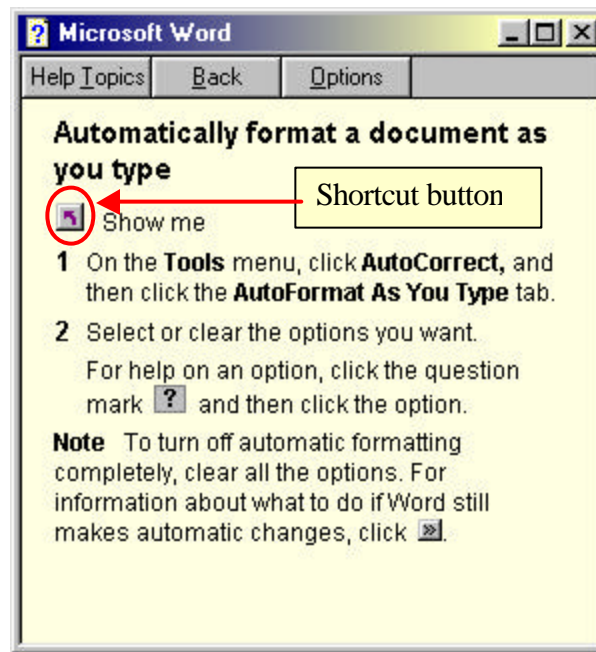
Show context to location.

Depending on the tool that you are using and whether the table of contents is viewable, ensure that the highlighted item in the table of contents changes whenever the user takes a hypertext jump to another topic in help. In this way, the users continually see the topic that they are consulting in the context of the table of contents.

Provide a home base with a quick return.

The home base is the user's entry point into the help system. It usually is the Help Topics window. Users should be able to return to that point from any other point in the help system. Providing a home button or menu button is an escape if a user is unable to retrace their steps or a shortcut if they are finished with the path they have been traveling.

Keep procedural Help window on top and to the right.

Procedural windows should appear on the right side of the screen in a small window. The size and position make it easy for users to carry out their tasks in the application while the Help topic is visible on the screen.

Don't state the obvious.

If the interface clearly shows what a particular item does, do not repeat this information in a procedure.

Include Related Topics only when a topic is closely related or is required for completing a task.

The Related Topics feature provides a key feature for a well-designed Help system. You should include entries for the most basic and useful information on the Contents page, then make less frequently used procedures and concepts available only through Related Topics. Define these topics to cover more in-depth information about a subject.

Use Wizards in Help with caution and for complex, low frequency tasks.

Wizards to provide help to the user are not well suited to teaching a user how to do something. Although wizards help the user accomplish a task, they are designed to hide many of the steps and much of the complexity of a given task. Use a wizard to supplement rather than replace the user's direct ability to perform a specific task. Do not rely on wizards as a solution for ineffective designs. If your users must rely too much on wizards, your application's interface might be overly complicated. See *The Windows User Experience* Book in the MSDN Online library for more information on designing wizards.

Always include information to tell user where to turn for more Help.

If Help cannot answer a question, direct the user to someone or something that can. Provide cross references to other online documents, recommend related paper documents, and give phone numbers of customer support and help desks. This can be seen below in the figure titled *Getting More Help*, where Microsoft has an extensive section on where the user can turn for Help on Microsoft Word.

Getting More Help
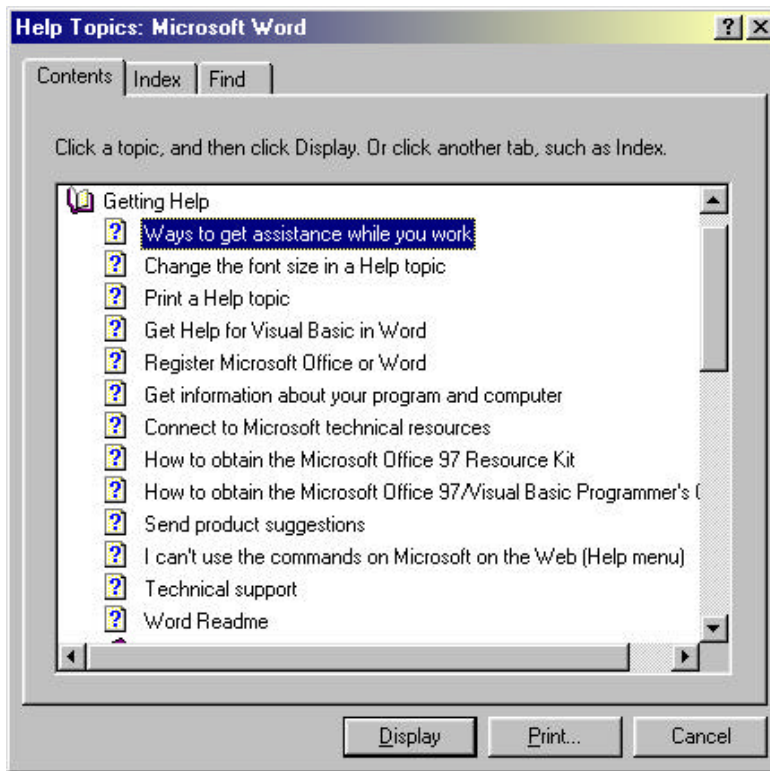

HTML Specific Help Standards

The following standards are to be adopted across all CDC web-based applications using HTML. These standards are intended to lead to a more consistent and usable interface.

Include Help in all CDC web-based applications and provide access to Help from the navbar.

Just because an application resides on the Web, doesn't mean it doesn't need good online Help. Just like Windows applications, users will have questions about how to perform tasks or how to solve problems. Even though there isn't a standard Help browser for the Web, all CDC surveillance application pages need to have access to Help from a menu or navbar as shown in the figure titled *Help on the Web* and should follow the guidelines above whenever possible. Until there is a standard Help browser for Web applications, the Help interface will vary somewhat. For example, supplying context-sensitive Help in an HTML application is somewhat difficult; therefore, developers may chose not to include this feature. However, as in Windows, strive for providing the user with the steps they need to get their work done as quickly as possible, and make the information easy for the user to find.

Help on the Web
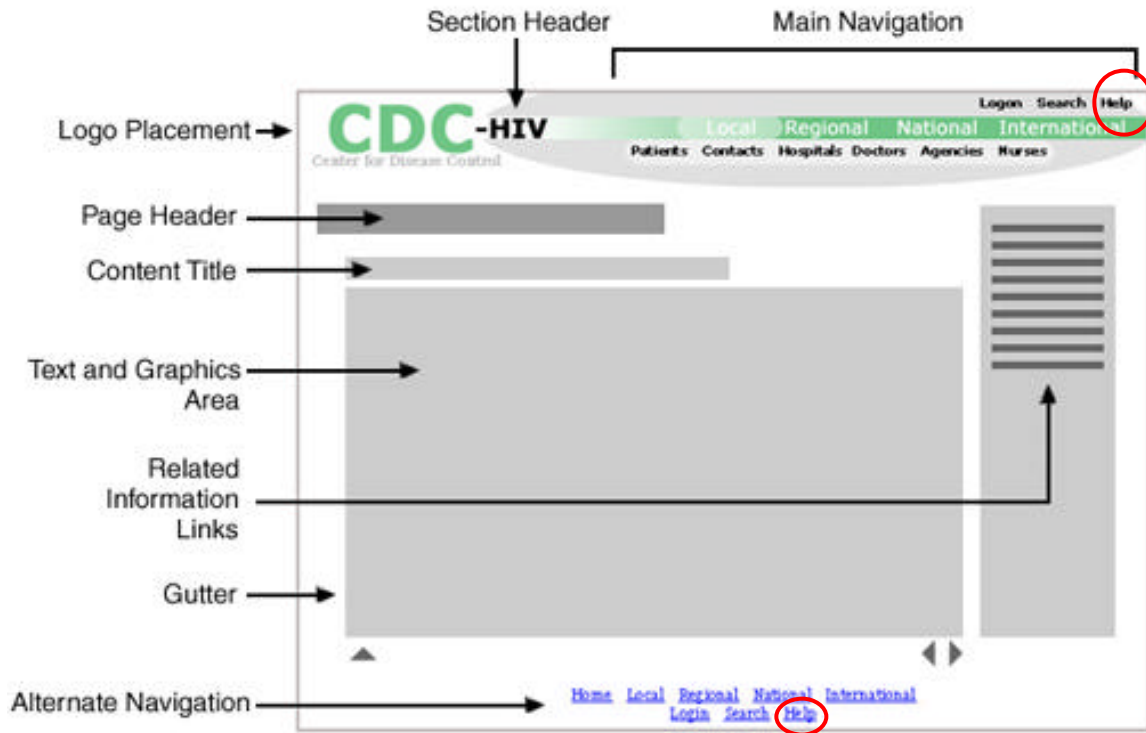
Recommended Readings

Horton, William. *Designing & Writing Online Documentation*, John Wiley & Sons, 1990
Chapter 10 - Help Facilities

McKay, Everett N. *Developing User Interfaces for Microsoft Windows.* Microsoft Press, 1999
Chapter 32 - Check Your Help System and Documentation

Windows 95 Help Authoring Kit. Microsoft Press, 1995
All chapters
Appendix A - Windows 95 Help Style Guidelines

The Windows Interface Guidelines for Software Design. Microsoft Press, 1995
Chapter 12 - User Assistance

The Windows User Experience. Microsoft Press, 1999
Chapter 13 - User Assistance

Helpful Web Sites

The Microsoft Developers Network Online Library of Books, specifically the online version of *The Windows Interface Guidelines for Software Design* now known as *The Windows User Experience.*
http://msdn.microsoft.com/isapi/msdnlib.idc?theURL=/library/books/winguide/PLATFRM2/D5/S115B5.HTM

Article on Deliverying documentation via the Internet
http://stc.org/region2/phi/meetings/meet0299.html